



(11) **EP 1 199 634 B1**

(12) **EUROPEAN PATENT SPECIFICATION**

(45) Date of publication and mention
of the grant of the patent:
09.04.2008 Bulletin 2008/15

(51) Int Cl.:
G06F 11/00 ^(2006.01) **G06F 11/20** ^(2006.01)

(21) Application number: **01307742.5**

(22) Date of filing: **12.09.2001**

(54) **Handling memory exhaustion in a data processing system**

Behandlung des Speicherüberlaufs in einem Datenverarbeitungssystem

Gestion d'épuisement de mémoire dans un système de traitement de données

(84) Designated Contracting States:
**AT BE CH CY DE DK ES FI FR GB GR IE IT LI LU
MC NL PT SE TR**

(30) Priority: **18.10.2000 US 691533**

(43) Date of publication of application:
24.04.2002 Bulletin 2002/17

(73) Proprietor: **International Business Machines
Corporation
Armonk, NY 10504 (US)**

(72) Inventors:
• **Bauman, Charles David,**
c/o IBM UK Ltd,
Int Prp Law
Winchester,
Hampshire SO21 2JN (GB)
• **Bealkowski, Richard,**
c/o IBM UK Ltd,
Int. Prop Law
Winchester,
Hampshire SO21 2JN (GB)

- **Clement, Thomas J.,**
c/o IBM UK Ltd,
Int Prop Law
Winchester,
Hampshire SO21 2JN (GB)
- **Pearce, Jerry William,**
c/o IBM UK Ltd,
Int Prp Law
Winchester,
Hampshire SO21 2JN (GB)
- **Turner, Michael R.,**
c/o IBM UK Ltd,
Int Prop Law
Winchester,
Hampshire SO21 2JN (GB)

(74) Representative: **Davies, Simon Robert**
IBM,
United Kingdom Limited,
Intellectual Property Law,
Hursley Park
Winchester, Hampshire SO21 2JN (GB)

(56) References cited:
EP-A- 0 726 514 **US-A- 5 511 177**
US-A- 5 574 952 **US-A- 5 974 506**

Note: Within nine months from the publication of the mention of the grant of the European patent, any person may give notice to the European Patent Office of opposition to the European patent granted. Notice of opposition shall be filed in a written reasoned statement. It shall not be deemed to have been filed until the opposition fee has been paid. (Art. 99(1) European Patent Convention).

EP 1 199 634 B1

Description

[0001] The present invention relates in general to data processing systems and in particular to data processing systems that support at least partial memory mirroring.

[0002] In the field of data processing systems, a design feature that can be employed in high-availability data processing systems is memory mirroring. Memory mirroring maintains a redundant copy of system memory. For example, if a system with two banks of random access memory (RAM) has memory mirroring, the first bank (e.g., Bank A) might be utilized as the primary bank, and the second bank (e.g., Bank B) might be utilized as a backup or mirror bank. The mirror bank serves as a repository for an extra copy of the data contained in the primary bank. In the event of errors in primary Bank A, the system avoids an interruption in service by utilizing mirror Bank B in lieu of primary Bank A. An error message may then be generated indicating that Bank A requires service.

[0003] A disadvantage associated with memory mirroring, however, is that less of the installed memory is available for general use. For example, in the above example, Bank A is available for general use, but Bank B is not. When all of the primary memory is mirrored onto mirror memory, the system is said to have full mirroring, in which case only half of the installed memory is available for general use. Alternatively, partial mirroring can be implemented. For example, a system with four banks of memory, (e.g., Banks A through D) may be configured to mirror Bank A onto Bank B but not to mirror Banks C and D. Partial mirroring provides for general use of a larger percentage of installed memory but less protection against memory errors.

[0004] Another design feature that can be employed to provide mirrored systems (as well as non-mirrored systems) with more apparent memory for general use is memory compression. A typical system with memory compression has a compression engine that utilizes the system's installed memory (i.e., physical memory) to support a larger amount of apparent memory (i.e., real memory). For example, if a system includes 32 MB of physical memory and a compression engine configured to provide a compression ratio of 2:1, the compression engine will report real memory of 64 MB. Consequently, if a system with 32 MB of installed memory has full mirroring and memory compression at a ratio of 2:1, the effects of the compression and mirroring offset each other. That is, the system will have a real memory size that equals its physical memory size.

[0005] A danger associated with memory compression, however, is that the configured compression ratio may not be obtainable for certain types of data. For example, if the data has already been compressed by software (e.g., if the data is a Joint Photographic Experts Group (JPEG) file), the compression engine may not be able to further compress the file enough to provide a 2:1 ratio. Furthermore, if the compression engine fails to

achieve the desired compression ratio and substantially all of the real memory is needed, the system may be unable to provide the required amount of real memory. When a system lacks sufficient physical memory to support all of the required real memory, the system is said to have suffered a condition known as memory exhaustion. However, memory exhaustion may also occur in systems without compression, for example if the system simply has insufficient physical memory to perform the required processing.

[0006] In conventional systems, exhausting physical memory is a critical system problem. That problem is typically signaled via an appropriately high priority event, such as a non-maskable interrupt (NMI) or a form of machine check. Consequently, in conventional systems, when memory compression and memory mirroring are used together, there will generally be a decreased risk of system failure due to memory errors but an increased risk of system failure due to memory exhaustion.

[0007] US Patent 5,511,177 discloses an alloc program which is called when a file system writes new data in a file. If there is no free data block in the volume, one mirror file in the volume is selected and deleted to generate a free block. This patent also discloses the creation of a mirror copy for frequently accessed files

[0008] Accordingly, the present invention provides systems, a method and computer program as defined in the appended claims. It will be appreciated that the system, method and computer program of the invention will typically share the same preferred features as one another.

[0009] In a preferred embodiment, a memory exhaustion condition is handled in a data processing system having first and second regions of physical memory. The memory exhaustion condition is detected while the second region is mirroring at least part of the first region. In response to the memory exhaustion condition, memory mirroring is at least partially deactivated. The memory exhaustion condition is then eliminated by augmenting the first region with at least part of the second region.

[0010] In an illustrative embodiment, the data processing system compresses real memory into the first region of physical memory, and the memory exhaustion condition is detected when the first region lacks sufficient available capacity to accommodate current requirements for real memory. The memory exhaustion condition is eliminated by compressing at least part of the real memory into the second region.

[0011] A preferred embodiment of the invention will now be described in detail by way of example only with reference to the following drawings:

Figures 1A and 1B depict block diagrams explaining the relationship between physical memory and real memory with mirroring enabled (in **Figure 1A**) and disabled (in **Figure 1B**) according to an illustrative embodiment of the present invention;

Figure 2 illustrates a block diagram of a data processing system representing an exemplary embodiment of the present invention, and having facilities for handling memory exhaustion conditions;

Figure 3 is a table illustrating settings of the memory-bank select bits for configuring mirroring in the system of **Figure 2**;

Figure 4 is a table depicting settings of a mirror-mode read bit for configuring mirroring in the system of **Figure 2**; and

Figure 5 illustrates a high-level logical flowchart of an exemplary process, according to the present invention, for handling memory exhaustion conditions.

[0012] With reference now to the figures, and in particular with reference to **Figures 1A** and **1B**, there are depicted two block diagrams explaining the relationship between physical memory and real memory in a data processing system with a memory subsystem that supports mirroring in accordance with an illustrative embodiment of the present invention. Specifically, the diagrams relate to a data processing system that has two banks of memory of equal capacity, Bank A **10** and Bank B **12**. In **Figure 1A**, the data processing system is configured in such a way that Bank A **10** serves as the primary bank and Bank B **12** serves as the mirror of Bank A **10**. In addition, the data processing system supports memory compression, such that the memory subsystem reports an address space for real memory **14** that exceeds the address space of the primary bank (*i.e.*, Bank A **10**).

[0013] To describe the relationship between primary memory and real memory more precisely, the following symbols are utilised:

X is the size of the physical, primary memory;
 X(m) is the size of the physical memory serving as a mirror;
 R is the compression ratio that the memory subsystem seeks to achieve (expressed as real-memory: primary-memory); and
 Rx is the size of the real memory, as reported by the memory subsystem.

In **Figure 1A**, R equals 2:1 and X equals X_1 ; consequently, Rx equals $2 * X_1$. Also, X(m) equals X_2 .

[0014] Referring now to **Figure 1B**, there is depicted a block diagram relating to the data processing system of **Figure 1A** with, however, the memory subsystem having been reconfigured so that Bank B **12** serves as primary memory, not mirror memory. In accordance with the present invention, such a reconfiguration may be performed automatically upon the occurrence of a memory exhaustion condition, as described in greater detail below. In the present illustrative embodiment, as a result of the reconfiguration, R equals 1:1 and X equals $X_1 + X_2$;

consequently, Rx equals $1 * (X_1 + X_2)$. Also, X(m) equals 0. Thus, additional physical memory is allocated to support needs for real memory, at the expense of loss of mirroring.

[0015] With reference now to **Figure 2**, there is illustrated a block diagram of a somewhat more complex exemplary data processing system **20** with facilities for handling memory exhaustion conditions in accordance with the present invention. As illustrated, data processing system **20** includes a system processor **22** that communicates with a memory subsystem **24** via a host bus **26**. Memory subsystem **24** includes a cache and memory controller (CMC) **30** which supports an L3 cache **32**. In addition, CMC **30** communicates with four banks of installed random access memory (RAM), Bank 0 **40**, Bank 1 **41**, Bank 2 **42**, and Bank 3 **43**, both directly and via a mirror mode controller (MMC) **50**. Other components (not illustrated) of data processing system **20** may communicate with memory subsystem **24** via a peripheral component interconnect (PCI) bus **52**.

[0016] MMC **50** contains a three-bit latch that can be written to and read by system processor **22** to alter the mirroring characteristics of memory subsystem **24**. The latch appears as a normal I/O port.

[0017] Referring now also to **Figure 3**, two bits of that latch are bank-select bits. By setting or clearing the bank-select bits to achieve the bit patterns illustrated in **Figure 3**, system processor **22** can alter how Bank 3 **43** is used. Bank 3 **43** is therefore said to have variable addressing. In particular, when at least one of bank-select bits 0 and 1 is cleared, Bank 3 **43** serves as a mirror bank for one of the primary banks. Such a configuration therefore implements partial-mirroring. In contrast, when bank-select bits 0 and 1 are both set, mirroring is deactivated, and Bank 3 **43** serves instead as another primary bank, for example as the high bank in a four bank array.

[0018] Referring now also to **Figure 4**, the third bit in the latch is a mirror-read bit which, when clear, causes memory reads to utilize a primary bank and, when set, causes memory reads to utilize the mirror bank in lieu of the corresponding primary bank. At this point, it may be noted that, if Bank 1 **41** and Bank 2 **42** were null and the capacities of Bank 0 **40** and Bank 3 **43** were equal, the system depicted in **Figure 2** would resemble or match the system described with reference to **Figures 1A** and **1B**.

[0019] The operation of memory subsystem **24** is governed by a memory manager **60** which, in the illustrative embodiment, is implemented as part of the operating system. Accordingly, memory manager **60** is loaded into RAM during the system boot process, along with other operating system components.

[0020] With reference now to **Figure 5**, there is illustrated a high-level logical flowchart of an exemplary process, according to the present invention, for handling memory exhaustion conditions within data processing system **20**. The process begins at block **70** with data processing system **20** being powered up. Then, as part

of the boot process, memory subsystem **24** is initialized, as depicted at block **72**. As part of the initialization process, CMC **30** determines how much physical memory is installed and memory manager **60** configures the bank-select and mirror-read bits within MMC **50** according to a default configuration. In the current exemplary process, MMC **50** will be configured for partial mirroring with Bank 0 **40** mirrored onto Bank 3 **43**.

[0021] A compression ratio may also be determined during this initialization process. In one embodiment, that ratio is determined based on historical compression performance, so that the initialized compression ratio may be automatically adjusted over time to more closely match the compression ratios that have actually been achieved during one or more prior operating cycles. In the present example, the compression ratio is set at 2:1.

[0022] After the memory subsystem has been initialized, data processing system **20** begins normal operation, as depicted at block **74**, with the memory subsystem operating as configured in the initialization process. During normal operation, combinational logic controls the generation of the chip select and output enable signals for each memory bank. This logic utilizes the high-order memory address lines, the bank-select bits, the system R/W line, and the mirror-read bit to generate the required signals and timing. In mirror mode, all writes to a mirrored primary bank (here, Bank 0 **40**) also write to the corresponding mirror bank (here, Bank 3 **43**). In addition, CMC **30** maintains usage tables for tasks such as allocating unused memory as needed and properly mapping real memory to physical memory.

[0023] During processing, CMC **30** utilizes L3 cache **32** to obtain recently accessed data, as data can be read from L3 cache **32** more quickly than from RAM. However, when the required data is not in L3 cache **32**, (*i.e.*, when there is a cache miss), it is necessary to access RAM. Specifically, in the present illustrative embodiment, a cache miss causes CMC **30** to read a compressed block of data from RAM, decompress the data, and load the data into the L3 cache. For a cache flush operation, the data to be written is compressed and written to RAM. The memory usage tables of CMC **30** are then updated accordingly.

[0024] As illustrated at block **76**, while system processor **22** executes instructions, memory manager **60** monitors memory subsystem **24** to detect whether a memory exhaustion condition exists (*i.e.*, whether the primary memory lacks sufficient available capacity to accommodate current requirements for real memory). For example, CMC **30** may be configured to generate a non-maskable interrupt or machine check when there simply is not enough physical memory to store the compressed contents of real memory, due to, for example, a request for additional real memory or a modification of existing memory contents. Alternatively, it may be determined that a memory exhaustion condition exists when current requirements for real memory cause the remaining capacity of unused physical memory to fall below a certain

threshold.

[0025] If no memory exhaustion condition is detected, processing continues with mirroring, as indicated by the arrow returning to block **74**. However, if a memory exhaustion condition is detected, the process passes to block **80**, which depicts memory manager **60** recording information pertaining to the memory exhaustion condition for use in adapting the configured compression ratio to align more closely with compression ratios that have actually been achieved. Specifically, in the current exemplary process, memory manager **60** computes a new compression ratio and stores that compression ratio in non-volatile storage. Acceptable storage media for this data include, without limitation, non-volatile (*e.g.*, battery-backed) RAM (*i.e.*, NVRAM), complementary metal-oxide semiconductor (CMOS) RAM, flash memory, and magnetic or optical disks. Other types of exhaustion information may also be stored.

[0026] After a memory exhaustion condition has been detected, memory manager **60** also reconfigures CMC **30** and MMC **50** (including setting both bank-select bits) to deactivate mirroring and convert Bank 3 **43** from a mirror bank to a primary bank, preferably as the high bank of a four-bank array, as depicted at blocks **82** and **84**. As a result, the amount of available primary memory is expanded, which eliminates the memory exhaustion condition. Then, as illustrated at block **86**, system processor **22** resumes normal operation, utilizing the expanded primary memory.

[0027] Also, as shown at block **88**, after normal processing has resumed, memory manager **60** monitors conditions to determine whether the expanded primary memory is still needed. If so, processing continues, as indicated by the arrow leading back to block **86**. However, if real memory requirements have subsided to the point that Bank 3 **43** is no longer needed as a primary bank, memory manager **60** reconfigures CMC **30** and MMC **50** to reactivate mirroring, as depicted at block **90**. Within the process of reactivating mirroring, the mirror bank is repopulated with data to match the corresponding primary bank. This repopulation can be performed by software or by hardware. System processor **22** then resumes normal operations, as indicated by the arrow leading back to block **74**.

[0028] As described herein therefore, memory exhaustion conditions can be handled by dynamically reconfiguring mirror banks to be primary banks. Mirroring is then reactivated when memory needs subside sufficiently. Furthermore, in data processing systems that provide memory compression, the startup compression ratio can be automatically adjusted based on how much compression has actually been realized in that data processing system.

[0029] It will be understood by those skilled in the art that various changes in form and detail may be made from the illustrative embodiments described so far. For example, full or partial mirroring may be utilised and different numbers of memory banks and different mirroring

proportions are supported as well, including data processing systems with N-way mirroring (*i.e.*, a data processing system with two or more mirror copies of data from one or more primary memory banks). In addition, the present invention can be utilized to advantage in systems with or without compression.

[0030] Another possibility is partial deactivation of mirroring to eliminate memory exhaustion conditions. For example, if a system is initially configured to utilize two or more banks of memory for mirroring, the system may respond to memory exhaustion conditions by reconfiguring one mirror bank as a primary bank and leaving mirroring active for the other mirror bank.

[0031] Although certain architectures have been utilized to illustrate exemplary embodiments, the present invention can be utilized in a very wide range of architectures, including, without limitation, general-purpose personal computers, multi-processor systems, mini-computers, and mainframe computers.

[0032] It will also be appreciated that although various operations of the present invention have been described as being performed by various components, including memory manager 60, CMC 30, and MMC 50, such operations could be distributed among these components differently. In addition, the implementation of certain components as hardware, software, and/or firmware can be altered according to particular system requirements. For example, memory manager 60 could be implemented partially or entirely in hardware and/or firmware. Likewise, rather than having a memory controller that generates signals such as interrupts to be processed by software, a data processing system with hardwired logic that controls mirroring may include hardwired detection logic that detects memory exhaustion conditions, as well as configuration logic that is hardwired to the detection logic in such a way that the configuration logic automatically disables mirroring and converts mirror banks into primary banks when necessary.

[0033] Furthermore, the present invention may be implemented as a computer program for use with a data processing system. Programs defining the functions described herein can be delivered to a data processing system via a variety of signal-bearing media, which include, without limitation, non-rewritable storage media (*e.g.*, CD-ROM), rewritable storage media (*e.g.*, a floppy diskette or hard disk drive), and communication media, such as digital and analog networks. It should be understood, therefore, that such signal-bearing media, when carrying or encoding computer readable instructions that direct these functions, represent alternative embodiments of the present invention.

Claims

1. A method of handling a memory exhaustion condition in a data processing system having first and second regions of physical memory, said method comprising:

prising:

detecting a memory exhaustion condition in said first region of physical memory while said second region is mirroring at least part of said first region (76);

the step of detecting a memory exhaustion condition comprising determining that said first region lacks sufficient available capacity to accommodate current requirements for real memory;

in response to said memory exhaustion condition, at least partially deactivating memory mirroring between said first and second regions (82);

augmenting said first region with at least part of said second region, such that said memory exhaustion condition is eliminated (86),

wherein the method is **characterised in that**:

the data processing system is operable to compress the first region such that the amount of reported real memory is greater than the physical memory of the first region in accordance with a configured compression ratio,

in response to said memory exhaustion condition, recording information pertaining to the memory exhaustion condition for use in adapting the configured compression ratio to align more closely with a compression ratio that has actually been achieved; and subsequently utilizing said recorded information to adjust the configured compression ratio.

2. The method of Claim 1, said data processing system having a mirror mode controller (50) that directs a memory access to said first region or said second region in accordance with one or more mirror mode bits, wherein:

said step of augmenting said first region comprises modifying at least one of said one or more mirror mode bits.

3. The method of Claim 1 or 2, further comprising:

after memory mirroring has been at least partially deactivated, determining that memory needs have diminished (88); and in response to determining that memory needs have diminished, at least partially reactivating mirroring (90).

4. A data processing system comprising:

first and second regions of physical memory; detection logic operable to detect a memory ex-

haustion condition in said first region of physical memory while said second region is mirroring at least part of said first region (76); and said detection logic comprises a memory controller operable to detect said memory exhaustion condition by determining that said first region lacks sufficient available capacity to accommodate current requirements for real memory; and configuration logic operable, responsive to said memory exhaustion condition, to at least partially deactivate memory mirroring between said first and second regions (82) and augment said first region with at least part of said second region, such that said memory exhaustion condition is eliminated (86),

wherein the data processing system is **characterised in that:**

the data processing is operable to compress the first region such that the amount of reported real memory is greater than the physical memory of the first region in accordance with a configured compression ratio; memory manager means, responsive to said memory exhaustion condition, for recording information pertaining to the memory exhaustion condition for use in adapting the configured compression ratio to align more closely with a compression ratio that has actually been achieved; and memory manager means for utilizing said information to adjust the configured compression ratio.

5. The data processing system of Claim 4, wherein:

said data processing system further comprises a mirror mode controller operable to direct a memory access to said first region or said second region in accordance with one or more mirror mode bits; and said configuration logic operable to augment said first region by modifying at least one of said one or more mirror mode bits.

6. A computer program comprising program instructions which when implemented on a data processing system perform the method of any of Claims 1 to 3.

7. A memory management system that handles a memory exhaustion condition in a data processing system having first and second regions of physical memory, said memory management system comprising:

detection logic operable to detect a memory exhaustion condition in said first region of physical

memory while said second region is mirroring at least part of said first region (76); and said detection logic comprises a memory controller operable to detect said memory exhaustion condition by determining that said first region lacks sufficient available capacity to accommodate current requirements for reported real memory, configuration logic operable, responsive to said memory exhaustion condition, to at least partially deactivate memory mirroring between said first and second regions (82) and augment said first region with at least part of said second region, such that said memory exhaustion condition is eliminated (86),

the system **characterised in that:**

wherein the data processing system is operable to compress the first region such that the amount of reported real memory is greater than the physical memory of the first region in accordance with a configured compression ratio, and memory manager means, responsive to said memory exhaustion condition, for recording information pertaining to the memory exhaustion condition for use in adapting the configured compression ratio to align more closely with a compression ratio that has actually been achieved; and memory manager means for utilizing said recorded information to adjust the configured compression ratio.

Patentansprüche

1. Verfahren zur Behandlung eines Speicherüberlaufzustands in einem Datenverarbeitungssystem, das über einen ersten und einen zweiten physischen Speicherbereich verfügt, wobei das Verfahren Folgendes umfasst:

Feststellen eines Speicherüberlaufzustands in dem ersten physischen Speicherbereich, während der zweite Bereich mindestens einen Teil des ersten Bereichs spiegelt (76);

wobei der Schritt des Feststellens eines Speicherüberlaufzustands die Feststellung umfasst, dass der erste Bereich nicht über ausreichend verfügbare Kapazität verfügt, um dem aktuellen Bedarf an realem Speicher Rechnung zu tragen; als Reaktion auf den Speicherüberlaufzustand mindestens teilweises Deaktivieren der Speicherspiegelung zwischen dem ersten Bereich und dem zweiten Bereich (82); Vergrößern des ersten Bereichs mit mindestens einem Teil des zweiten Bereichs, so dass der Speicherüberlaufzustand beseitigt wird (86),

wobei das Verfahren **dadurch gekennzeichnet ist, dass:**

das Datenverarbeitungssystem in der Lage ist, den ersten Bereich so zu komprimieren, dass das Volumen an gemeldetem realem Speicher entsprechend einem konfigurierten Komprimierungsverhältnis größer als der physische Speicher des ersten Bereichs ist; als Reaktion auf den Speicherüberlaufzustand Informationen erfasst werden, die sich auf den Speicherüberlaufzustand beziehen und bei der Anpassung des konfigurierten Komprimierungsverhältnisses zu verwenden sind, um eine höhere Übereinstimmung mit einem Komprimierungsverhältnis zu erzielen, das tatsächlich erreicht worden ist; und die erfassten Informationen anschließend verwendet werden, um das konfigurierte Komprimierungsverhältnis anzupassen.

2. Verfahren nach Anspruch 1, wobei das Datenverarbeitungssystem über eine Steuereinheit (50) für den Spiegelmodus verfügt, die einen Speicherzugriff auf den ersten Bereich oder den zweiten Bereich entsprechend einem oder mehreren Spiegelmodus-Bits steuert, wobei:

der Schritt des Vergrößerns des ersten Bereichs die Änderung von mindestens einem des einen oder der mehreren Spiegelmodus-Bits umfasst.

3. Verfahren nach Anspruch 1 oder 2, das des weiteren Folgendes umfasst:

nachdem die Spiegelung des Speichers mindestens teilweise deaktiviert worden ist, Ermitteln, ob der Speicherbedarf zurückgegangen ist (88); und als Reaktion auf die Feststellung, dass der Speicherbedarf zurückgegangen ist, mindestens teilweises Reaktivieren der Spiegelung (90).

4. Datenverarbeitungssystem, das Folgendes umfasst:

einen ersten und einen zweiten physischen Speicherbereich; Erkennungslogik, die in der Lage ist, einen Speicherüberlaufzustand in dem ersten physischen Speicherbereich zu erkennen, während der zweite Bereich mindestens einen Teil des ersten Bereichs spiegelt (76); und

wobei die Erkennungslogik eine Speichersteuereinheit umfasst, die in der Lage ist, den Speicherüberlaufzustand zu erkennen, indem sie feststellt, dass der erste Bereich nicht über ausreichend verfügbare

Kapazität verfügt, um dem aktuellen Bedarf an realem Speicher Rechnung zu tragen; und Konfigurationslogik, die in der Lage ist, als Reaktion auf den Speicherüberlaufzustand die Speicherspiegelung zwischen dem ersten Bereich und dem zweiten Bereich mindestens teilweise zu deaktivieren (82) und den ersten Bereich mit mindestens einem Teil des zweiten Bereichs zu vergrößern, so dass der Speicherüberlaufzustand beseitigt wird (86); wobei das Datenverarbeitungssystem **dadurch gekennzeichnet ist, dass:**

das Datenverarbeitungssystem in der Lage ist, den ersten Bereich so zu komprimieren, dass das Volumen an gemeldetem realem Speicher entsprechend einem konfigurierten Komprimierungsverhältnis größer als der physische Speicher des ersten Bereichs ist; ein Speicherverwaltungsmittel, das auf den Speicherüberlaufzustand anspricht, um Informationen zu erfassen, die sich auf den Speicherüberlaufzustand beziehen und bei der Anpassung des konfigurierten Komprimierungsverhältnisses zu verwenden sind, um eine höhere Übereinstimmung mit einem Komprimierungsverhältnis zu erzielen, das tatsächlich erreicht worden ist; und ein Speicherverwaltungsmittel, um die Informationen zur Anpassung des konfigurierten Komprimierungsverhältnisses zu verwenden.

5. Datenverarbeitungssystem nach Anspruch 4, wobei:

das Datenverarbeitungssystem des Weiteren eine Steuereinheit für den Spiegelmodus umfasst, die in der Lage ist, einen Speicherzugriff auf den ersten Bereich oder den zweiten Bereich gemäß einem oder mehreren Spiegelmodus-Bits zu steuern; und die Konfigurationslogik in der Lage ist, den ersten Bereich zu vergrößern, indem sie mindestens eines des einen oder der mehreren Spiegelmodus-Bits ändert.

6. Rechnerprogramm, das Programmbefehle umfasst, die, wenn sie in einem Datenverarbeitungssystem umgesetzt werden, das Verfahren nach einem der Ansprüche 1 bis 3 durchführen.

7. Speicherverwaltungssystem, das einen Speicherüberlaufzustand in einem Datenverarbeitungssystem behandelt, welches über einen ersten und einen zweiten physischen Speicherbereich verfügt, wobei das Speicherverwaltungssystem Folgendes umfasst:

Erkennungslogik, die in der Lage ist, einen Spei-

cherüberlaufzustand in dem ersten physischen Speicherbereich zu erkennen, während der zweite Bereich mindestens einen Teil des ersten Bereichs spiegelt (76); und

wobei die Erkennungslogik eine Speichersteuereinheit umfasst, die in der Lage ist, den Speicherüberlaufzustand zu erkennen, indem sie feststellt, dass der erste Bereich nicht über ausreichend verfügbare Kapazität verfügt, um dem aktuellen Bedarf an gemeldetem realem Speicher Rechnung zu tragen, wobei das Datenverarbeitungssystem in der Lage ist, den ersten Bereich zu komprimieren, so dass das Volumen an gemeldetem realem Speicher gemäß einem konfigurierten Komprimierungsverhältnis größer als der physische Speicher des ersten Bereichs ist, und Konfigurationslogik, die in der Lage ist, als Reaktion auf den Speicherüberlaufzustand die Speicherspiegelung zwischen dem ersten Bereich und dem zweiten Bereich mindestens teilweise zu deaktivieren (82) und den ersten Bereich mit mindestens einem Teil des zweiten Bereichs zu vergrößern, so dass der Speicherüberlaufzustand beseitigt wird (86); wobei das System **gekennzeichnet ist durch:**

ein Speicherverwaltungsmittel, das auf den Speicherüberlaufzustand anspricht, um Informationen zu erfassen, die sich auf den Speicherüberlaufzustand beziehen und bei der Anpassung des konfigurierten Komprimierungsverhältnisses zu verwenden sind, um eine höhere Übereinstimmung mit einem Komprimierungsverhältnis zu erzielen, das tatsächlich erreicht worden ist; und ein Speicherverwaltungsmittel, um die erfassten Informationen zur Anpassung des konfigurierten Komprimierungsverhältnisses zu verwenden.

Revendications

1. Procédé de gestion d'une condition d'épuisement de mémoire dans un système de traitement de données ayant des première et deuxième régions de mémoire physique, ledit procédé comprenant :

la détection d'une condition d'épuisement de mémoire dans ladite première région de mémoire physique alors que ladite deuxième région est en train de créer une image miroir d'au moins une partie de ladite première région (16), l'étape de détection d'une condition d'épuisement de mémoire comprenant la détermination du fait que ladite première région manque d'une capacité disponible suffisante pour prendre en charge les exigences actuelles de mémoire réel-

le, en réponse à ladite condition d'épuisement de mémoire, la désactivation au moins partielle de la création d'image miroir de mémoire entre les première et deuxième régions (82), l'augmentation de ladite première région avec au moins une partie de ladite deuxième région, de sorte que ladite condition d'épuisement de mémoire est éliminée (86),

où le procédé est **caractérisé en ce que :**

le système de traitement de données peut être mis en oeuvre pour compresser la première région de sorte que la quantité de mémoire réelle signalée est supérieure à la mémoire physique de la première région, conformément à un taux de compression configuré, en réponse à ladite condition d'épuisement de mémoire, l'enregistrement d'informations se rapportant à la condition d'épuisement de mémoire pour une utilisation pour adapter le taux de compression configuré pour s'aligner plus étroitement avec un taux de compression qui a réellement été obtenu, et l'utilisation ultérieure desdites informations enregistrées pour ajuster le taux de compression configuré.

2. Procédé selon la revendication 1, ledit système de traitement comportant un contrôleur de mode de miroir (50) qui dirige un accès à la mémoire vers ladite première région ou ladite deuxième région en fonction d'un ou plusieurs bits de mode de miroir, dans lequel :

ladite étape d'augmentation de ladite première région comprend la modification d'au moins un desdits un ou plusieurs bits de mode miroir.

3. Procédé selon la revendication 1 ou 2, comprenant en outre :

après que la création d'image miroir de mémoire a été au moins partiellement désactivée, la détermination du fait que les besoins de mémoire ont diminué (88), et en réponse à la détermination du fait que les besoins de mémoire ont diminué, la réactivation au moins partielle de la création d'image miroir (90).

4. Système de traitement de données, comprenant :

des première et deuxième régions de mémoire physique, une logique de détection pouvant être mise en oeuvre pour détecter une condition d'épuise-

ment de mémoire dans ladite première région de mémoire physique alors que ladite deuxième région crée une image miroir d'au moins une partie de ladite première région (76), et ladite logique de détection comprend un contrôleur de mémoire pouvant être mis en oeuvre pour détecter ladite condition d'épuisement de mémoire en déterminant que ladite première région manque d'une capacité disponible suffisante pour prendre en charge des exigences actuelles de mémoire réelle, et une logique de configuration pouvant être mise en oeuvre, en réponse à ladite condition d'épuisement de mémoire, pour désactiver au moins partiellement la création d'image miroir de mémoire entre lesdites première et deuxième régions (82) et augmenter ladite première région avec au moins une partie de ladite deuxième région, de sorte que ladite condition d'épuisement de mémoire soit éliminée (86),

où le système de traitement de données est **caractérisé en ce que** :

le traitement de données peut être mis en oeuvre pour compresser la première région de sorte que la quantité de mémoire réelle signalée est supérieure à la mémoire physique de la première région conformément à un taux de compression configuré, un moyen de gestion de mémoire, répondant à ladite condition d'épuisement de mémoire, destiné à enregistrer des informations se rapportant à la condition d'épuisement de mémoire pour une utilisation pour adapter le taux de compression configuré pour s'aligner plus étroitement avec un taux de compression qui a réellement été obtenu, et un moyen de gestion de mémoire destiné à utiliser lesdites informations pour ajuster le taux de compression configuré.

5. Système de traitement de données selon la revendication 4, dans lequel :

ledit système de traitement de données comprend en outre un contrôleur de mode de miroir pouvant être mis en oeuvre pour diriger un accès à la mémoire vers ladite première région ou ladite deuxième région conformément à un ou plusieurs bits de mode de miroir, et ladite logique de configuration pouvant être mise en oeuvre pour augmenter ladite première région en modifiant au moins l'un desdits un ou plusieurs bits de mode de miroir.

6. Programme informatique comprenant des instructions de programme qui, lorsqu'elles sont mises en

oeuvre sur un système de traitement de données, exécutent le procédé selon l'une quelconque des revendications 1 à 3.

7. Système de gestion de mémoire qui gère une condition d'épuisement de mémoire dans un système de traitement de données comportant des première et deuxième régions de mémoire physique, ledit système de gestion de mémoire comprenant :

une logique de détection pouvant être mise en oeuvre pour détecter une condition d'épuisement de mémoire dans ladite première région de mémoire physique alors que ladite deuxième région est en train de créer une image miroir d'au moins une partie de ladite première région (76), et

ladite logique de détection comprend un contrôleur de mémoire pouvant être mis en oeuvre pour détecter ladite condition d'épuisement de mémoire en déterminant que ladite première région manque d'une capacité disponible suffisante pour prendre en charge les exigences actuelles de mémoire réelle signalée

une logique de configuration pouvant être mise en oeuvre en réponse à ladite condition d'épuisement de mémoire, pour désactiver au moins partiellement la création d'image miroir de mémoire entre lesdites première et deuxième régions (82) et augmenter ladite région avec au moins une partie de ladite deuxième région, de sorte que ladite condition d'épuisement de mémoire est éliminée (86),

le système étant **caractérisé en ce que** :

le système de traitement de données peut être mis en oeuvre pour compresser la première région de sorte que la quantité de mémoire réelle signalée est supérieure à la mémoire physique de la première région conformément à un taux de compression configuré, et

un moyen de gestion de mémoire, répondant à ladite condition d'épuisement de mémoire, destiné à enregistrer des informations se rapportant à la condition d'épuisement de mémoire pour une utilisation pour adapter le taux de compression configuré pour s'aligner plus étroitement avec un taux de compression qui a été réellement obtenu, et

un moyen de gestion de mémoire destiné à utiliser lesdites informations enregistrées pour ajuster le taux de compression configuré.

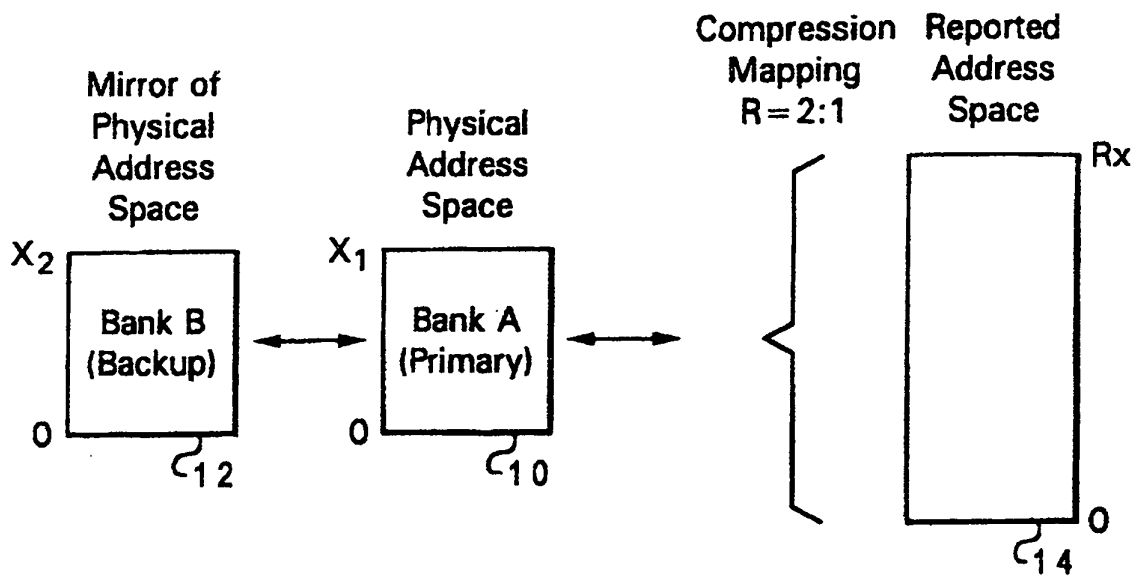


Fig. 1A

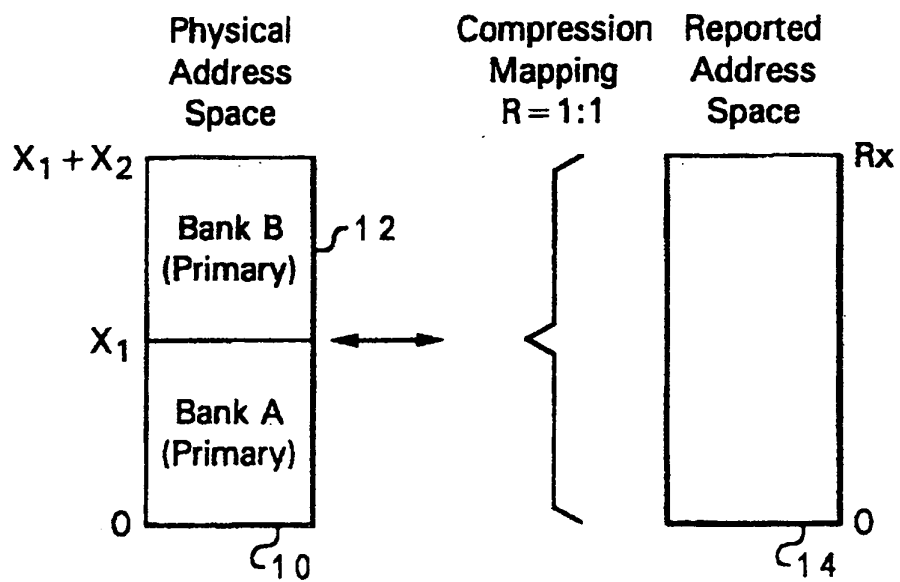


Fig. 1B

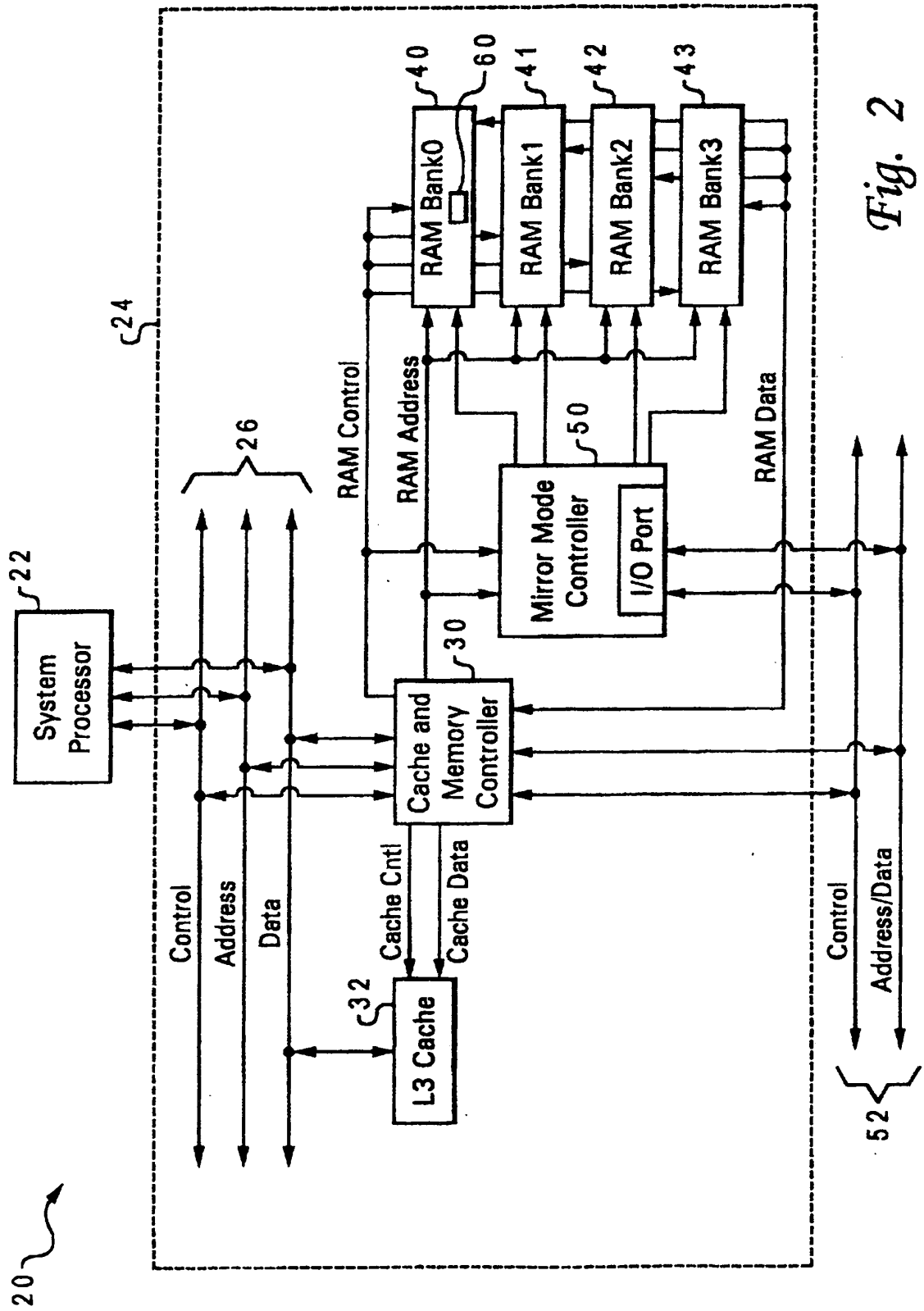


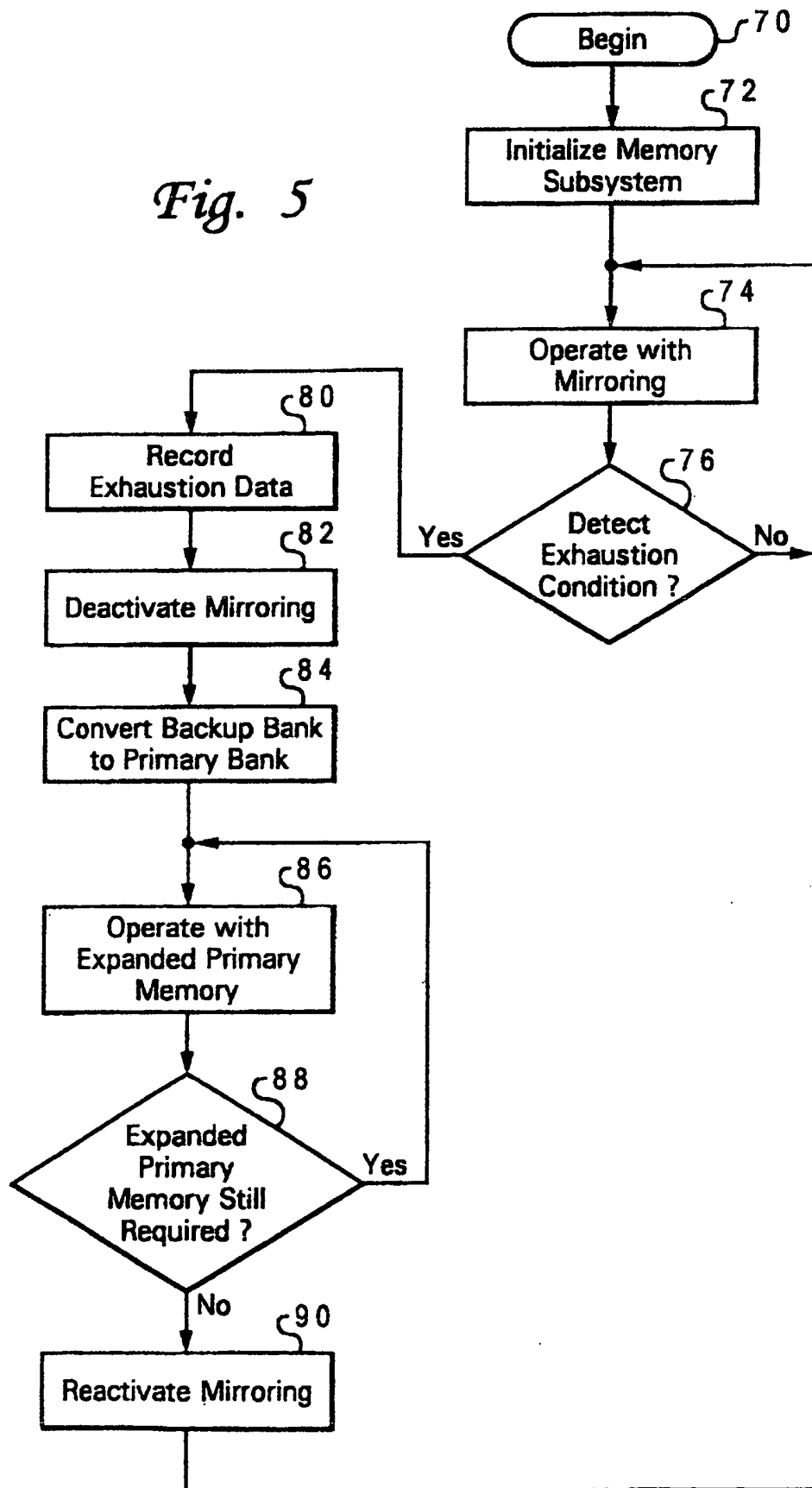
Fig. 2

BSEL1	BSEL0	
0	0	Mirror Bank0 on Bank3
0	1	Mirror Bank1 on Bank3
1	0	Mirror Bank2 on Bank3
1	1	Use Bank3 as the high bank of a 4-bank array

Fig. 3

MR	
0	Read from primary bank
1	Read from Bank3 in lieu of primary bank

Fig. 4

Fig. 5

REFERENCES CITED IN THE DESCRIPTION

This list of references cited by the applicant is for the reader's convenience only. It does not form part of the European patent document. Even though great care has been taken in compiling the references, errors or omissions cannot be excluded and the EPO disclaims all liability in this regard.

Patent documents cited in the description

- US 5511177 A [0007]